

MULTI 5.3.x Integrated Development Build Options Guide

Supports Cafe SDK

Version 1.03

Advanced Data Controls Corp.



Copyright 2011-2013 Green Hills Software, Inc. & Advanced Data Controls Corporation. All rights reserved.

**The content of this document is highly confidential
and should be handled accordingly.**

Table of Contents

1	Build Options	3
1.1	Build Options Set with Cafe SDK	3
1.2	Options for Floating-Point Operation Results	7
1.3	Optimization Options	8
	Revision History	10

1 Build Options

This chapter describes the build options for the MULTI Integrated Development Environment from Green Hills Software.

1.1 Build Options Set with Cafe SDK

In Cafe SDK, the Green Hills Software build options are set in the following makefile.

```
cafe_sdk\system\src\build\make\commondefs.cos.gh.mk
```

This chapter describes the default build options.

Build Options	Description	Category
-cpu=espresso	Specifies the CPU. Do not change it.	Advanced Targets
-I <directory>	Specifies the directory the compiler searches for header files.	Project
-dbg_source_root <directory>	Specifies the source root used by the MULTI debugger's <code>sourceroot</code> command. Set as the root directory for the project during debugging. (There is normally no need to specify this.)	Project
-Onone -Ogeneral (-O) -Osize -Ospeed	-Onone Disallows all optimizations. (The default optimizations cannot be disabled, however.) -Ogeneral or -O Handles optimizations that improve both execution speed and size. -Osize or -Ospace Allows all optimizations that improve both size and execution speed and special optimizations that improve the code size at the expense of execution speed. -Ospeed Handles optimizations that improve both execution speed and size.	Optimization
-OI	Allows all general optimization options to use 2-passes/inlining. However, 2-passes/inlining is not enabled in the Cafe make environment because files are compiled individually. When this option is specified, inlining is handled on a per-file basis.	Optimization

Build Options	Description	Category
-Olink	Enables linker optimizations. Avoid specifying this option routinely for large projects because it significantly increases the linking time.	Optimization
-Owholeprogram	Enables optimizations for the entire program. However, optimization of the program overall is not enabled in the Cafe make environment because files are compiled individually.	Optimization
-Omax	Executes the maximum optimization without consideration for the compiling time. Combines the <code>-Osize</code> and <code>-Ospeed</code> optimization options.	Optimization
-G	Generates source-level debugging information. Enables procedure calls from the MULTI debugger command line.	Debugging
-dual_debug	Generates debugging information in the object file in DWARF v2 format as well as Green Hills' <code>.dbo</code> format. (Necessary with using third-party debuggers other than MULTI.)	Advanced debugging → Native debugging information
-sda -zda	Puts a fixed amount of data in the SDA region or ZDA region to make memory access more efficient. <code>-sda</code> Enables SDA optimization, sets the threshold value to 8 (bytes). <code>-sda=size</code> Enables SDA optimizations. Specifies the threshold value (in bytes) of objects to put into SDA. SDA optimization is not done if you specify none. <code>-zda=size</code> Enables ZDA optimizations. Specifies the threshold value (in bytes) of objects to put into ZDA. <code>-nothreshold</code> Disables SDA/ZDA optimizations.	Target → Text and data layout
--exceptions --no_exceptions	<code>--exceptions</code> Allows support for processing C++ exceptions. <code>--no_exceptions [default]</code> Disallows support for processing C++ exceptions.	C/C++ Compiler
--link_once_templates	Enables the link-once method for generating instances of non-exportable templates.	C/C++ Compiler

Build Options	Description	Category
-no_ansi_alias	Disallows ANSI alias rules. To allow ANSI aliases, specify -ansi_alias.	C/C++ Compiler
--quit_after_warnings	Quits the build on a build error when warnings are generated. To prevent the build from quitting on warnings, specify --no_quit_after_warnings.	Compiler Diagnostics
-no_commons	Do not treat uninitialized global variables as common data. (An extern declaration is required for external variable references.)	C/C++ compiler/C++ data allocation
-C99 -c99 -ANSI -ansi -gcc (C compiler language specification)	-C99 Compiles using the strict ISO C99 specification (ISO/IEC 9899:1999). -c99 Compiles using the ISO C99 specification, but some non-standard features result in warnings. (Normally specify this option in the Cafe environment.) -ANSI Compiles using the strict ANSI C specification (ANSI X3.159-1989). -ansi Compiles using the ANSI C specification (ANSI X3.159-1989), but some non-standard features result in warnings. -gcc Compiles in GNU mode.	C/C++ Compiler

Build Options	Description	Category
--STD --std --g++ --e --ee (C++ compiler language specification)	--STD Compiles using the strict standard C specification (ISO/IEC 14882:1998). --std Compiles using the Standard C++ specification mode, but some non-standard features result in warnings. --g++ Specifies the GNU C++ mode as the C++ language specification. (Normally specify this option in the Cafe environment.) --e Compiles using the embedded C++ mode. --ee Compiles using the extended embedded C++ mode.	C/C++ Compiler
-kanji=XXXXX	-kanji=shiftjis Specifies the parsing for <i>kanji</i> that use the Shift-JIS encoding on the host or target. -kanji=utf8 Specifies UTF-8 for multibyte encoding and UCS-2 for wide-character encoding. -kanji=shiftjis/utf8 Specifies the parsing for <i>kanji</i> that use Shift-JIS encoding on the host and UTF-8 for multibyte encoding or UCS-2 for wide-character encoding on the target.	C/C++ compiler → Special token
-only_explicit_reg_use	Does not use register implicitly when copying structures.	Advanced Targets
-preprocess_assembly_files	Preprocesses assembly source files that have standard filename extensions.	Assembler

1.2 Options for Floating-Point Operation Results

The following build options can affect floating-point operation results. These options are all disabled by default. Use them with caution. Also, for performance reasons, enabling these options by default is not recommended.

Build Options	Description	Category
-precise_signed_zero	Uses something other than the IEEE floating-point standard for signed zeros (-0.0 and +0.0).	Floating point
-fNaN_cmp_unordered	Prohibits conversions to compilers where NaN would render IEEE floating-point operation results invalid. For example, $(x \leq y)$ does not become the same as $!(x > y)$.	Floating point
-no_use_fsel	Does not use the <code>fsel</code> instructions, which result in inaccurate comparisons between a NaN and infinity with the same sign.	Target

1.3 Optimization Options

The table below shows the various optimizations and the optimization options that enable or disable them.

Option	Command	Optimization Option				
		-Onone	-Odebug	-Osize	-O	-Ospeed
Inline tiny functions	--inline_tiny_functions --no_inline_tiny_functions	OFF	OFF	ON (if -G/-g is not specified)		
Inline C memory functions	-Omemfuncs -Onomemfuncs	OFF	OFF	OFF	ON	ON
Inline C string functions	-Ostrfuncs -Onostrfuncs	OFF	OFF	OFF	ON	ON
Simplifies C print functions.	-Oprintfuncs -Onoprintfuncs	OFF	OFF	ON	ON	ON
Loop/unroll	-Ounroll -Onounroll	OFF	OFF	OFF	OFF	ON
Assign register by coloring	-overload -nooverload	ON	ON	ON	ON	ON
Auto-assign to register	-autoregister -noautoregister	ON	ON	ON	ON	ON
Partially erase common area	-Ocse -Onocse	OFF	OFF	ON	ON	ON
Tail recursion	-Otailrecursion -Notailrecursion	OFF	OFF	ON	ON	ON
Constant propogation	-Oconstprop -Onoconstprop	OFF	OFF	ON	ON	ON
C/C++ Minimum/Maximum values	-Ominmax -Onominmax	OFF	OFF	ON	ON	ON
Memory optimization	-OM -Onomemory	OFF	ON	ON	ON	ON
Loop optimization	-OL -Onoloop	OFF	OFF	OFF	OFF	ON

Option	Command	Optimization Option				
Peephole optimization	-Opeep -Onopeephole	OFF	OFF	ON	ON	ON
Pipeline instruction scheduling	-Opipeline -Onopipeline	OFF	OFF	ON	ON	ON
Peeling during vectorization	-Ovectorpeel -Onovectorpeel	ON	ON	ON	ON	ON

Revision History

Revision	Description
1.03	Initial version.

All company and product names in this document are the trademarks or registered trademarks of their respective companies.

